

**Федеральное государственное автономное образовательное
учреждение высшего образования «Санкт-Петербургский
национальный исследовательский университет
информационных технологий, механики и оптики»**

Кафедра Вычислительной Техники

Лабораторная работа №3 Вычислительная математика

«Численное интегрирование»

Вариант : 6

Выполнил : Каримов Фаррух

Группа : Р3202

Цель работы: найти приближенное значение определенного интеграла с требуемой точностью различными численными методами.

Исходные данные:

1. Пользователь выбирает функцию, интеграл которой требуется вычислить (3-5 функций), из тех, которые предлагает программа.
2. Пределы интегрирования задаются пользователем.
3. Точность вычисления - 10^{-6} .
4. Начальное значение числа разбиения интервала интегрирования – $n=4$.

Программная реализация задачи:

1. Реализовать в программе методы по выбору пользователя:
 - * Метод прямоугольников (3 модификации: левые, правые, средние)
 - * Метод трапеций
 - * Метод Симпсона
2. Предусмотреть вывод результатов: значение интеграла, число разбиения интервала интегрирования для достижения требуемой точности.

Вывод программы :

```
Hello user!
We are going to work with :
f1(x) = x + 5.; f2(x) = x^2; f3(x) = cos(x + 3.) + 17.; f4(x) = e^x - 17.;
Write function id :
function id = 2
Write interval of integration (A; B):
A = 1
B = 2
Choose one of ways to solve
Algorithms : rectanglesLeft(1), rectanglesRight(2), rectanglesMiddle(3), trapezoidMethod(4), SimpsonMethod(5)
write algorithm number (1-5).
1
k = 19,0000000
n = 1048576
Integral = 2,3333326
Dou you want continue ? ( yes/no )
нуы
yes
We are going to work with :
f1(x) = x + 5.; f2(x) = x^2; f3(x) = cos(x + 3.) + 17.; f4(x) = e^x - 17.;
Write function id :
function id = 2
Write interval of integration (A; B):
A = 1
B = 2
Choose one of ways to solve
Algorithms : rectanglesLeft(1), rectanglesRight(2), rectanglesMiddle(3), trapezoidMethod(4), SimpsonMethod(5)
write algorithm number (1-5).
2
k = 19,0000000
n = 1048576
Integral = 2,3333340
Dou you want continue ? ( yes/no )
yes
We are going to work with :
f1(x) = x + 5.; f2(x) = x^2; f3(x) = cos(x + 3.) + 17.; f4(x) = e^x - 17.;
Write function id :
function id = 2
Write interval of integration (A; B):
A = 1
B = 2
Choose one of ways to solve
```

```

Algorithms : rectanglesLeft(1), rectanglesRight(2), rectanglesMiddle(3), trapezoidMethod(4), SimpsonMethod(5)
write algorithm number (1-5).
3
k = 7,0000000
n = 256
Integral = 2,3333330
Dou you want continue ? ( yes/no )
yes
We are going to work with :
f1(x) = x + 5.; f2(x) = x^2; f3(x) = cos(x + 3.) + 17.; f4(x) = e^x - 17.;
Write function id :
function id = 2
Write interval of integration (A; B):
A = 1
B = 2
Choose one of ways to solve
Algorithms : rectanglesLeft(1), rectanglesRight(2), rectanglesMiddle(3), trapezoidMethod(4), SimpsonMethod(5)
write algorithm number (1-5).
4
k = 8,0000000
n = 512
Integral = 2,3333335
Dou you want continue ? ( yes/no )
yes
We are going to work with :
f1(x) = x + 5.; f2(x) = x^2; f3(x) = cos(x + 3.) + 17.; f4(x) = e^x - 17.;
Write function id :
function id = 2
Write interval of integration (A; B):
A = 1
B = 2
Choose one of ways to solve
Algorithms : rectanglesLeft(1), rectanglesRight(2), rectanglesMiddle(3), trapezoidMethod(4), SimpsonMethod(5)
write algorithm number (1-5).
5
k = 1,0000000
n = 4
Integral = 2,3333333
Dou you want continue ? ( yes/no )
no
Bye. Happy new year.

```

Сама программа :

```

• public class Main {
public static void main(String[] args) {
    Scanner scanner = new Scanner(System.in);
    PrintWriter printWriter = new PrintWriter(System.out);
    MyLog.println(printWriter, "Hello user!");
    solve(scanner, printWriter);
    MyLog.println(printWriter, "Bye. Happy new year.");
}
public static void solve(Scanner scanner, PrintWriter printWriter){
    boolean flag = true;
    while (flag){
        MyLog.println(printWriter, "We are going to work with : ");
        MyLog.println(printWriter, Functions.printFuncs());
        MyLog.println(printWriter, "Write function id : ");
        MyLog.print(printWriter, "function id = ");
        int functionId = scanner.nextInt();
        while (!(1 <= functionId && (functionId <= 4))){

```

```

        MyLog.println(printWriter, "Wrong function id. Please choose one from 1 - 4.");
    }
    MyLog.println(printWriter, "Write interval of integration (A; B): ");
    MyLog.print(printWriter, " A = ");
    double a = scanner.nextDouble();
    MyLog.print(printWriter, " B = ");
    double b = scanner.nextDouble();
    MyLog.println(printWriter, "Choose one of ways to solve ");
    MyLog.println(printWriter, " Algorithms : rectanglesLeft(1), rectanglesRight(2), rectanglesMiddle(3), " +
        "trapezoidMethod(4), SimpsonMethod(5)");
    MyLog.println(printWriter, " write algorithm number (1-5). ");
    int algorithmId = scanner.nextInt();
    while(!((1 <= algorithmId) || (algorithmId <= 5))){
        MyLog.println(printWriter, "Wrong number. Please write algorithm number, number in a range 1 - 5 . ");
        algorithmId = scanner.nextInt();
    }
    Solvers.solverGo(printWriter, algorithmId, functionId, a, b);
    MyLog.println(printWriter, " Dou you want continue ? ( yes/no ) ");
    String userAns = scanner.next();
    while(!((userAns.equals("yes")) || (userAns.equals("no")))){
        userAns = scanner.next();
    }
    if(userAns.equals("no")) flag = false;
}
}
}

```

- **public class MyLog {**
public static void print(PrintWriter printWriter, String s){
 printWriter.print(s);
 printWriter.flush();
 }
public static void println(PrintWriter printWriter, String s){
 printWriter.println(s);
 printWriter.flush();
 }
public static void print(PrintWriter printWriter, double d){
 printWriter.printf("%.7f", d);
 printWriter.flush();
 }
public static void println(PrintWriter printWriter, double d){
 printWriter.printf("%.7f \n", d);
 printWriter.flush();
 }
 }

- **public class Solvers {**
public static void solverGo(PrintWriter printWriter, int solverId, int functionId, double a, double b){
 int n = 4;
 double eps = 0.000001;
 double I0 = useSolver(solverId, functionId, n, a, b);
 double I1 = useSolver(solverId, functionId, 2 * n, a, b);
 int k = 1;
 while(Math.abs(I1 - I0) > eps){
 k++;
 n *= 2;
 I0 = I1;
 I1 = useSolver(solverId, functionId, 2 * n, a, b);
 }
 MyLog.print(printWriter, " k = ");
 MyLog.println(printWriter, k);
 MyLog.print(printWriter, " n = ");
 }

```

MyLog.println(printWriter, Integer.toString(n));
MyLog.print(printWriter, " Integral = ");
MyLog.println(printWriter, l1);
}

private static double useSolver(int solverId, int functionId, int n, double a, double b){
    switch (solverId){
        case 1 : return rectanglesLeft(functionId, n, a, b);
        case 2 : return rectanglesRight(functionId, n, a, b);
        case 3 : return rectanglesMiddle(functionId, n, a, b);
        case 4 : return trapezoidMethod(functionId, n, a, b);
        default: return SimpsonMethod(functionId, n, a, b);
    }
}

public static double rectanglesLeft(int functionId, int n, double a, double b){
    double sum = 0.;
    double h = (b - a) / (n * 1.);
    for(int i = 0; i < n; i++){
        sum += Functions.getFunc(functionId, a + h * (i * 1.));
    }
    sum *= h;
    return sum;
}

public static double rectanglesRight(int functionId, int n, double a, double b){
    double sum = 0.;
    double h = (b - a) / (n * 1.);
    for(int i = 1; i <= n; i++){
        sum += Functions.getFunc(functionId, a + h * (i * 1.));
    }
    sum *= h;
    return sum;
}

public static double rectanglesMiddle(int functionId, int n, double a, double b){
    double sum = 0.;
    double h = (b - a) / (n * 1.);
    for(int i = 1; i <= n; i++){
        double x = ((a + h * (i * 1.)) + (a + h * ((i - 1) * 1.))) / 2.;
        //double x = a + (h * (i * 1.)) - (h / 2.);
        sum += Functions.getFunc(functionId, x);
    }
    sum *= h;
    return sum;
}

public static double trapezoidMethod(int functionId, int n, double a, double b){
    double sum = 0.;
    double h = (b - a) / (n * 1.);
    for(int i = 1; i < n; i++){
        sum += Functions.getFunc(functionId, a + h * (i * 1.));
    }
    sum *= 2.;
    sum += Functions.getFunc(functionId, a) + Functions.getFunc(functionId, b);
    sum = (sum * h) / 2.;
    return sum;
}

public static double SimpsonMethod(int functionId, int n, Double a, Double b){
    Double sum = Functions.getFunc(functionId, a) + Functions.getFunc(functionId, b);
    Double h = (b - a) / (n * 1.);
    for(int i = 1; i < n; i++){

```

```

    if(i % 2 == 1) sum += 4. * Functions.getFunc(functionId, a + h * (i * 1.));
    else sum += 2. * Functions.getFunc(functionId, a + h * (i * 1.));
}
sum = (sum * h) / 3.;
return sum;
}
}

```

- **public class Functions {**

```

    public static double getFunc(int id, double x){
        switch (id){
            case 1 : return func1(x);
            case 2 : return func2(x);
            case 3 : return func3(x);
            default: return func4(x);
        }
    }
    public static String printFuncs(){
        return "f1(x) = x + 5.; f2(x) = x^2; f3(x) = cos(x + 3.) + 17.; f4(x) = e^x - 17.;";
    }
    public static double func1(double x){
        return x + 5.;
    }
    public static double func2(double x){
        return Math.pow(x, 2);
    }
    public static double func3(double x){
        return Math.cos(x + 3.) + 17.;
    }
    public static double func4(double x) {
        return Math.pow(Math.E, x) - 17.;
    }
}

```

Вывод : в ходе выполнения данной лабораторной работы, я изучил различные способы численного интегрирования. Применил на практике эти методы, и получил практический опыт.